

Chapter 11 - Linear Regression and Model Building

Supplemental Code

Brian Fogarty

This document provides the code for creating the second scatterplot in the *Ordinary Least Squares Process* subsection (in the Bivariate Linear Regression section), the scatterplot in the *Confidence Intervals* subsection (in the Bivariate Linear Regression section), and an interactive three-dimensional plot for multiple linear regression (mentioned in a footnote but not included the chapter).

OLS Process Scatterplot

```
library(tidyverse)
simd <- read_csv("simd2020.csv", na = "")

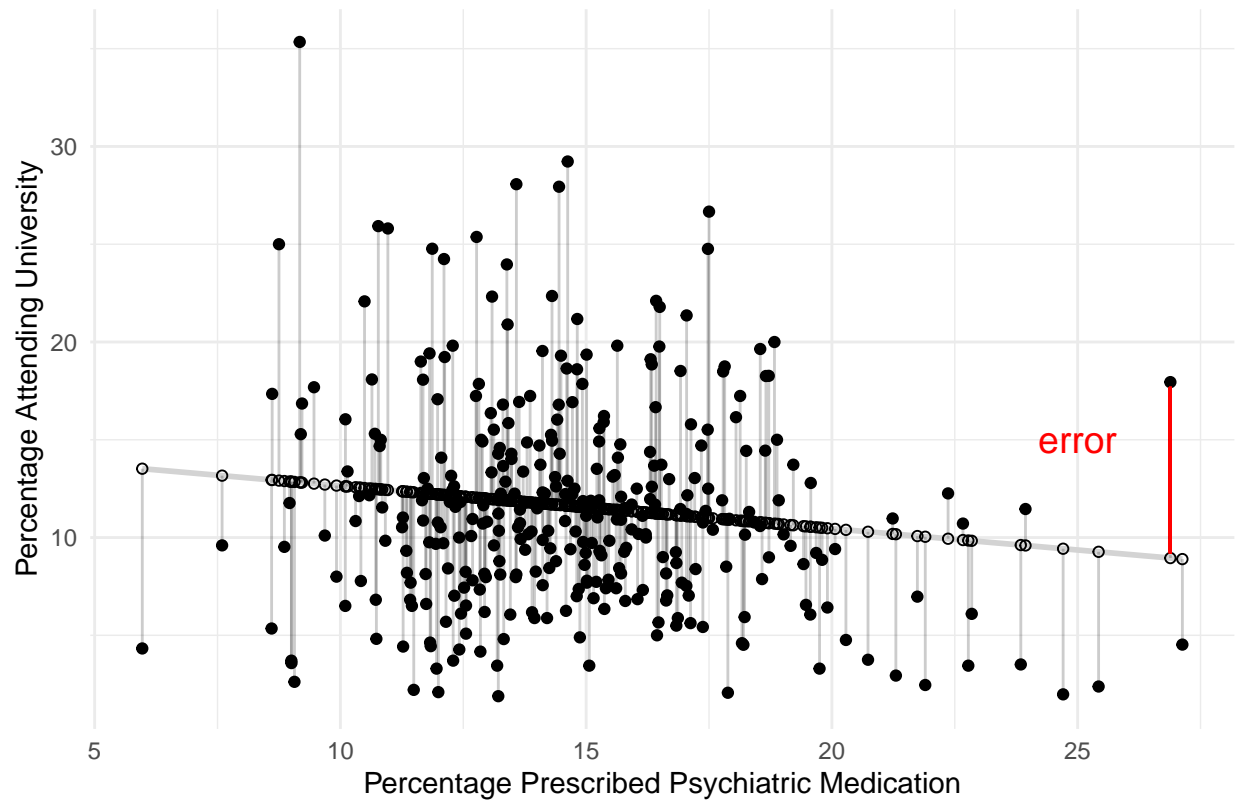
test <- simd %>%
  rename_with(tolower) %>%
  mutate(pct_depress = depress*100,
         pct_uni = university*100,
         pct_broadband = broadband*100) %>%
  select(pct_uni, pct_broadband, pct_depress) %>%
  filter(pct_broadband >= 50)

model.test <- lm(pct_uni ~ pct_depress, data = test)

test$predicted <- predict(model.test)
test$residuals <- residuals(model.test)

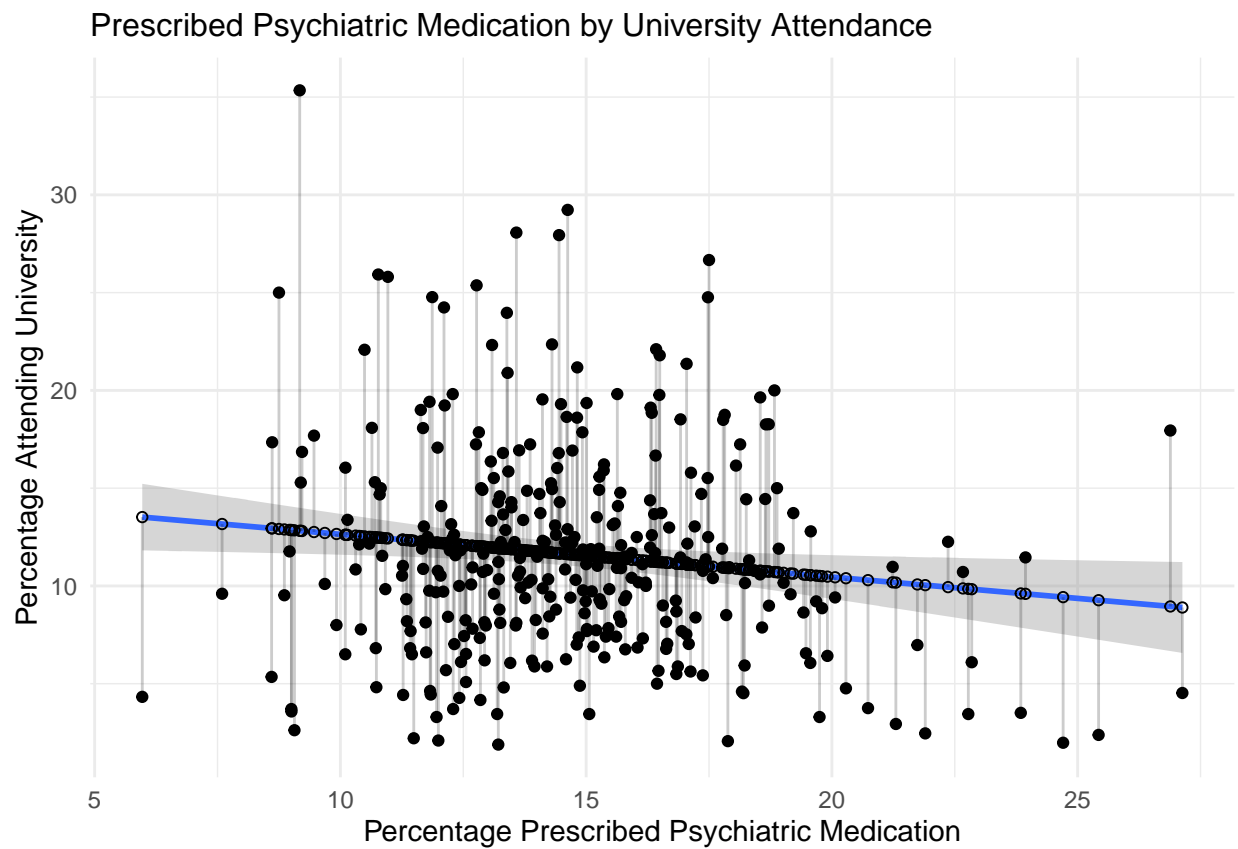
ggplot(test, aes(x = pct_depress, y = pct_uni)) +
  geom_smooth(method = "lm", se = FALSE, color = "lightgrey") +
  geom_segment(aes(xend = pct_depress, yend = predicted), alpha = .2) +
  geom_point() +
  geom_point(aes(y = predicted), shape = 1) +
  theme_minimal() +
  theme(
    plot.title = element_text(size = 12)
  ) +
  labs(x = "Percentage Prescribed Psychiatric Medication",
       y = "Percentage Attending University",
       title = "Prescribed Psychiatric Medication by University Attendance") +
  geom_curve(aes(x = 26.88679, y = 17.7, xend = 26.88679, yend = 9.2),
            colour = "red", curvature = 0) +
  annotate("text", label = "error", x = 25, y = 15, colour = "red", size = 5)
```

Prescribed Psychiatric Medication by University Attendance



Scatterplots with Confidence Intervals

```
ggplot(test, aes(x = pct_depress, y = pct_uni)) +  
  geom_smooth(method = "lm", se = TRUE) +  
  geom_segment(aes(xend = pct_depress, yend = predicted), alpha = .2) +  
  geom_point() +  
  geom_point(aes(y = predicted), shape = 1) +  
  theme_minimal() +  
  theme(  
    plot.title = element_text(size = 12)  
  ) +  
  labs(x = "Percentage Prescribed Psychiatric Medication",  
       y = "Percentage Attending University",  
       title = "Prescribed Psychiatric Medication by University Attendance")
```



Three-Dimensional Plot for OLS

This code is adapted from Ismay, Chester, and Albert Y. Kim. 2022. *Statistical Inference via Data Science: A Modern Dive into R and the Tidyverse*. CRC Press. https://github.com/moderndive/ModernDive_book/blob/master/06-multiple-regression.Rmd

I've included `eval=FALSE` in the chunk header so the code is not run in the pdf knitting. If you run the below code, an interactive three-dimensional plot will appear in the **Viewer** window in RStudio.

```
library(plotly)

test1 <- simd %>%
  rename_with(tolower) %>%
  mutate(pct_depress = depress*100,
         pct_uni = university*100,
         pct_broadband = broadband*100,
         pct_employment_deprived = employment_rate*100) %>%
  select(pct_uni, pct_broadband, pct_depress, pct_employment_deprived) %>%
  filter(pct_broadband >= 50, pct_employment_deprived <= 20)

model.test1 <- lm(pct_uni ~ pct_depress + pct_employment_deprived, data = test1)

x_grid <- seq(from = min(test1$pct_depress),
              to = max(test1$pct_depress), length = 100)
y_grid <- seq(from = min(test1$pct_employment_deprived),
              to = max(test1$pct_employment_deprived), length = 100)
z_grid <- expand.grid(x_grid, y_grid) %>%
  as_tibble() %>%
  rename(x_grid = Var1, y_grid = Var2) %>%
  mutate(z = coef(model.test1)[1] + coef(model.test1)[2] * x_grid +
         coef(model.test1)[3] * y_grid) %>%
  .[["z"]] %>%
  matrix(nrow = length(x_grid)) %>%
  t()

plot_ly() %>%
  add_markers(
    x = test1$pct_depress,
    y = test1$pct_employment_deprived,
    z = test1$pct_uni,
    hoverinfo = "text",
    text = ~ paste(
      "Pct Depress",
      test1$pct_depress,
      "</br> Pct Employment Deprived",
      test1$pct_employment_deprived,
      "</br> Pct Uni",
      test1$pct_uni
    )
  ) %>%
  layout(
    scene = list(
      xaxis = list(title = "Pct Depress"),
      yaxis = list(title = "Pct Employment Deprived"),
```

```
    zaxis = list(title = "Pct Uni")
  )
) %>%
add_surface(
  x = x_grid,
  y = y_grid,
  z = z_grid
)
```